Preprint
UCRL-JC-137651

Real-Time Local Range On-Demand and Dynamic Regional Range Images

L.V. Tsap

This article was submitted to
Institute of Electrical and Electronics Engineers
Workshop on Human Modeling, Analysis and Synthesis
Hilton Head, SC
June 13-16, 2000

U.S. Department of Energy

Lawrence
Livermore
National
Laboratory

February 22, 2000

Real-Time Local Range On-Demand and Dynamic Regional Range Images

Leonid V. Tsap
CASC

Abstract [1]

This paper presents a new approach to a gesture tracking system using real-time range on-demand. The system represents a gesture-controlled interface for interactive visual exploration of large data sets. The paper describes a method performing range processing only when necessary and where necessary. Range data is processed only for non-static regions of interest. This is accomplished by a set of filters on the color, motion, and range data. The speedup achieved is between 41% and 54%. The algorithm also includes a robust skin color segmentation insensitive to illumination changes. Selective range processing results in dynamic regional range images (DRRIs). This development is also placed in a broader context of a biological visual system emulation, specifically redundancies and attention mechanisms.

1 Introduction

Recent years have seen a drastic increase in the size and complexity of scientific data. NIH's Visible Human project generated data sets of a single 3-D volume consisting of 12 billion elements. Nearly a terabyte of satellite data is produced daily. Advanced physics simulation here, at the Lawrence Livermore National Laboratory (LLNL), is responsible for generating large data sets, which plan to increase to one terabyte every five minutes by the year 2004. Traditional visualization represents the last step in data processing. However, efficiency of such processing suffers when errors are discovered at this point, and the entire data analysis cycle has to start over. Therefore, the trend in data growth is amplified by increasing requirements for interactive data access, display, exploration, analysis and collaboration. Focused on the development of efficient techniques addressing these requirements, SAVAnTS (Scalable Algorithms for Visualization and Analysis of Terascale Science) project is a collaboration between the Center for Applied Scientific Computing at LLNL and multiple academic partners. With such substantial amounts of data to explore, we are also interested in developing new interactive settings that would allow scientists to explore their data in a more intuitive environment. The data would be projected on a large screen, and updated in real-time following gesture-based commands of interacting scientists. The gesture tracking system described in this paper will be responsible for supplying data manipulation parameters to interactive data exploration and collaborative visualization software, and to virtual reality systems.

Since the system is developed as a front end for gesture-controlled large-scale visualization and virtual reality manipulation, certain requirements and complications are obvious. First, 3-D information is required, not necessarily at a video frame rate, but at least a few times per second (optimal parameters should be determined as a result of testing on a large group of people). Second, not only arms or hands, but also the entire body of the interacting person is moving. More over, interaction will take place in front of the large screen where the data being manipulated will be displayed. Most of the time the data will be updated dynamically as a result of such manipulations, and, therefore, traditional techniques such as background subtraction cannot easily separate figure from the background. Third, motion of the interacting person should be natural and result in intuitive data manipulation, where intuitive means easy to learn and fast to achieve immediate results.

Object tracking from image sequences is a very important research domain. Goals of object tracking include segmenting each frame into differently moving objects, selecting the object of interest, and analyzing its motion during the entire sequence or multiple sequences. Therefore, object tracking involves processing of both spatial and temporal data. A number of applications is dealing with tracking the motion of the human body. These applications include videosurveillance, gesture-based interfaces to multimedia applications and systems, interfaces for people with disabilities that prevent them from using the standard input technology, and videoconferencing. The most popular mode of HCI is based on devices like keyboards and mice, which limit the speed and naturalness of the interaction [12]. There is a continuing effort to involve human communication through movement in the design and development of computer interfaces that adequately capture such natural forms

[1] This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract number W-7405-Eng-48. UCRL-JC-136053

of communication. Another application is object manipulation in virtual environments.

Traditional approaches to tracking typically relied on segmentation of the intensity data, using motion or appearance data. A majority of the methods began by segmenting the human body from the background. For instance, in "blob approaches" people were modeled as a number of blobs resulting from pixel classification based on their color and position in the image. Wren *et al.* [19] achieved segmentation by classifying pixels into one of several models, including a static world and a dynamic user represented by gaussian blobs. Yang and Ahuja [21] used skin color and the geometry of palm and face regions for segmentation stages of their system. A Gaussian mixture (with parameters estimated by an EM algorithm) modeled the distribution of skin color pixels. Regh and Kanade [14] used a 3-D hand model to track a hand. They compared line features from the images with the projected model and performed incremental state corrections. Similar work was presented by Kuch and Huang [10] in which the synthesis process could fit the hand model to any person's hand. Bobick and Wilson [3] treated gesture as a sequence of states and computed configuration states along prototype gestures. Yacoob and Black proposed parameterized representation of human movement [20]. Cutler and Davis [5] segmented the motion and computed a moving objects self-similarity (including human motion experiments). A review by Aggarwal and Cai [1] classified approaches to human motion analysis, the tasks involved, and major areas related to human motion interpretation. A review by Pavlovic *et al.* [12] addressed main components and directions in gesture recognition research for HCI.

It is known that color-based skin detection techniques are susceptible to variability in lighting conditions [12]. Some common solutions included [12]: specially colored gloves or markers, restrictive backgrounds or clothing, prior knowledge of initial hand positions, or movement restrictions. Goals of our project exclude such simplifications. Instead we use the SCT/Center algorithm that can handle changing illumination. It was originally developed for skin cancer detection using color features [18]. Later the algorithm was successfully tested for position estimation of micro-rovers [13].

Usefulness of 3-D data in gesture analysis applications is not questionable. Since most machine vision system try to recover useful information about a scene from its projections, having three-dimensional (3-D) data eliminates ambiguities in solving the inversion of a many-to-one mapping. The projection of human movement often can be affected by the observation viewpoint and the distance from the camera [20]. Most gesture tracking and recognition applications could certainly benefit from including range data and having more information recovered from a scene. However, until recently, using range data for tracking was not feasible because of the speed and cost considerations. Some authors used multiple cameras and models to obtain 3-D locations of body parts. Azarbayejani and Pentland [2] triangulated on blobs composing a model. Gavrila and Davis [7] addressed whole-body tracking with four cameras placed in the corners of the room. Segen and Kumar [15] used depth cues from projections of the hand and its shadow for 3-D hand pose estimation. Otherwise range data was used in motion analysis primarily in an offline mode [16, 17].

Recent availability of less expensive, faster range data makes it a feasible additional source of information for tracking. This is the first real-time gesture tracking system that utilizes on-demand range in both spatial and temporal representations. It will be applied to natural navigation and visualization of large data sets. The method is also applicable to virtual reality systems. Oda *et al.* [11] reported application of a real-time range to virtual reality which utilized comparison of the depth information in real and synthetic data. In addition to the efficient range processing, proposed method also deals with the major shortcoming of color-based localization methodologies variability of the skin color classification results under different illumination conditions.

2 Description of the Method

Both color and range image are *grabbed synchronously*, and color image is extracted and rectified (corrected for lens distortions). However, *range is not processed* at this point (Figure 1) as one would expect. Instead, a number of filters are applied to the color data. These filters achieve a goal of localizing regions of interest (ROIs), specifically hands for our application (since their motion will provide input to visualization programs).

First, color feature filters are applied. The spherical coordinate transform (SCT) separates the color and brightness information. Color normalization provides SCTs insensitivity to variations in illumination (see Appendix 2). LAB space is computed, and pixels are classified as skin are computed using derived statistical data. A skin classifier with minimum distance classifier using Mahalanobis distance (see Appendix 2) selects pixels that can be considered skin pixels.

Noise removal is achieved with a sequence of erosions and dilations. The connected component analysis is performed next by scanning from left to right and from top to bottom, labeling, and evaluating equivalencies. Resulting regions are sorted, and small regions are removed from further consideration. Region area is evaluated with respect to the image size. Other geometry and shape filters are designed to eliminate re-

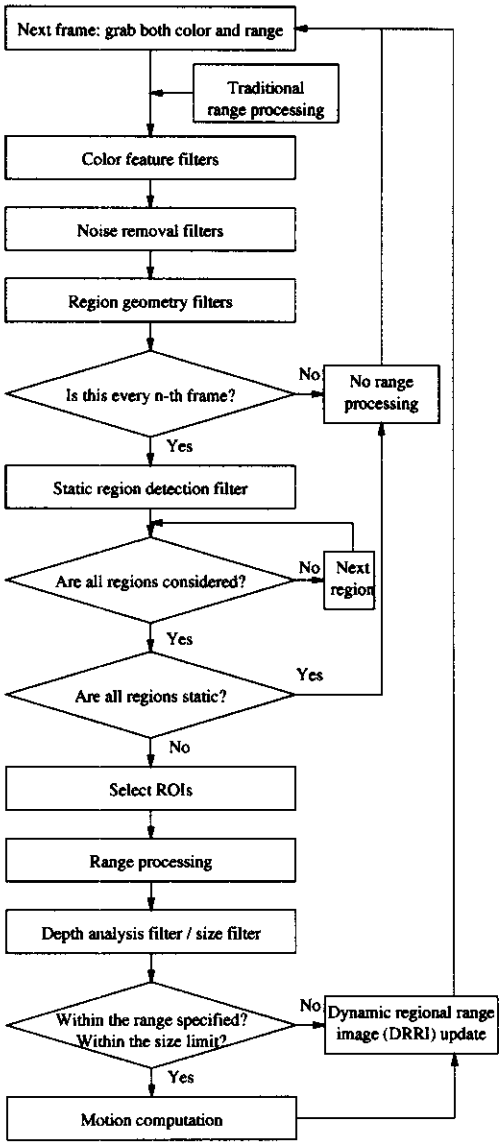


Figure 1: Algorithm of the range on-demand approach.

gions with unlikely shapes for human faces or hands, including long regions, regions with very few pixels (less than 30 %) classified as skin.

The human hands and faces are difficult to detect when only color information is used. Experiments in the next section use a simulation of a possible virtual scene with objects of various shapes and sizes, and a robotic hand which could be following human gestures in a completed back-end virtual reality application. One of the objects, a ball in the center of the scene, is given color properties very similar to human skin to confuse the program.

However, additional filters are set to prevent such confusion. A static region detection filter (defined for frames after the first) determines the absence of current motion for a given region. The filter evaluates it proportionally to the average noise level and the image size (since motion considered neglectable for relatively large images can be considered important for smaller ones). The process results in dynamic regional range images (DRRIs). Static regions are shown on DRRIs as outlines only, since range is not computed for them. DRRIs contain range information for regions of interest (with pixels still classified as skin after color- and geometry-based filters) moving in the current frame, outlines for static regions and recent motion information for both.

Only non-static regions are selected for range processing which takes place at this point (again let us note that color and range were grabbed synchronously, only the range processing was postponed). Stereo is estimated only for selected ROIs. Thus, the computation bottleneck is greatly reduced (see next section for speedup percentages vs. region sizes).

Next, the depth analysis filter evaluates whether ROIs exhibit face or hands geometry since the depth is known at this step, absolute sizes are computed. Non-human regions are excluded from the motion computation, but currently still tracked on the color images and DRRIs for visual purposes. If skin colored moving objects pass previous filters, then they are unveiled at this point (and not included into motion computation). The entire algorithm of the range on-demand approach is shown in Figure 1.

3 Experimental Results

The following experiments involve application of the algorithm to color and range image sequences of gestures. Triclops color stereo vision system (manufactured by Point Grey Research, Vancouver, Canada) is used to capture these sequences. The module connects to a single-processor Pentium III PC. Range information is recovered in real time from a correlation-based trinocular stereo algorithm (see Appendix 1 for details about the algorithm).

Typical color and range images produced by the stereo vision system are shown in Figure 2. Closer objects appear lighter in the range data, except for the darkest areas (for instance, some hair and far wall regions) indicating that no correspondence was found during the stereo matching process. As a result of applying color information-based filters, skin regions are selected (shown as white areas in the binary image in Figure 3(a), and as a rectangular enclosing boxes in Figure 3(b)).

Selected frames from sequences of intensity and DRR images are shown in Figures 5 and 4. Frames are chosen when both color and range information was scheduled to be processed (every $n-th$ frame in the algorithm, where n=5). To tell the preparation stage from the nucleus and retraction stages, interactors will use a fist as if grabbing the object being manipulated. That is why frames, taken during ges-

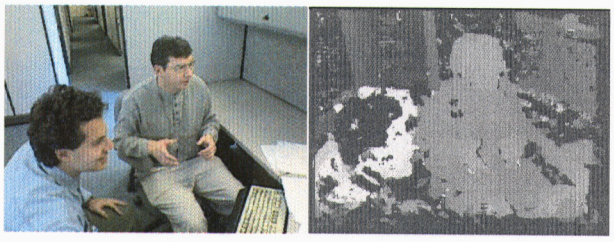


Figure 2: Typical color and range images produced by the stereo vision system.

(a) (b)

Figure 3: Pixels classified as skin as a result of applying color information-based filters.

tures signifying object operations, show persons using closed fists. Since manipulation of virtual objects is one of popular applications of hand gestures [12], the background represents a scene with virtual objects and a robotic arm-manipulator, one of them (a ball) is skin-colored. Of course, the main application of the system, as discussed in the Introduction, will be interactive exploration of visualized large data sets.

The first sequence shows tracking of a zoom gesture (hand moving towards the camera). First frame shows that all three candidate skin regions are detected: the face, the hand, and the virtual ball (created as a distracter with color similar to skin). These regions pass color feature filters, noise removal and region geometry filters, and static region detection filters (since relative motion is not defined for the first frame). However, the ball is not tracked beyond the first frame since it is obviously a static object. Note that the tracking system developed is a front-end for the interactive visualization software, and, therefore, background subtraction is not the best option in the general case. No apparently static regions are processed (the face and the ball). Otherwise, they would have been excluded from motion analysis on the basis of depth or size or both by the last filters. Hand motion was detected in frames 3, 7 and 10, and reflected in respective DRRIs.

Similarly, the second sequence has hand motion in six frames (not considering the first one) and head motion in two frames (8 and 9). Obviously, keeping ones head completely motionless is not a practical consideration, and head motion is present in all frames. In most frames, however, it does not pass the small motion filter (based on the average noise level and the image size).

DRRIs can be included in motion analysis, trajectory computation, gesture recognition, to determine what types of gestures are natural and feasible for robust tracking and interpretation for interactive exploration of large data sets and virtual environments. They can be easily plotted in a 3-D space for movement trajectory parameterization. Also they can produce (also in 3-D) templates for recognition of movements somewhat similar to the temporal templates [6].

Tables 1 and 2 contain statistics for corresponding motion sequences. Frame numbers correspond to respective frames in figures. "Number of ROIs" column indicates regions selected for range processing, next column contains their total area. Percentage of total image size is also included, as well as the total time for this frame (for the entire algorithm to process it) and the speedup over an average non-ROI processing time per frame (488 ms).

Average frame rate for longer sequences is also measured and averaged. Processing 1 range image for every 4 color images is done at a rate of 10.6 frames per second for a 320x240 image size, and at a rate of 16.5 frames per second for 160x120 images. Therefore, the method is applicable to the real-time processing. Moreover, since the Triclops library is currently optimized for thread parallel processing, a much greater speedup can be achieved on a dual-processor NT machine (we plan to move the system there in the near future).

Table 1: Statistics for the forward hand motion.

Frame (Fig. 4)	Number of ROIs	Total area of ROIs	% of area size	Time for this frame, ms	Speedup, %
1	3	5105	6.65	281	42.5
2	0	0	0	240	50.8
3	1	2030	2.64	261	46.6
4	0	0	0	233	52.2
5	0	0	0	227	53.4
6	0	0	0	233	52.2
7	1	3268	4.26	274	43.9
8	0	0	0	233	52.2
9	0	0	0	234	52.1
10	1	4992	6.50	280	42.6

Table 2: Statistics for the side-to-side hand motion.

Frame (Fig. 5)	Number of ROIs	Total area of ROIs	% of area size	Time for this frame, ms	Speedup, %
1	3	8690	11.32	287	41.1
2	0	0	0	240	50.8
3	1	690	0.90	267	45.3
4	1	837	1.09	267	45.3
5	1	837	1.09	267	45.3
6	0	0	0	233	52.2
7	1	868	1.13	267	45.3
8	2	6719	8.75	281	42.5
9	2	6425	8.37	280	42.6
10	0	0	0	241	50.7

The speedup is significant (between 41% and 54%). However, according to equation 3 (see Appendix 1), the speedup per frame should be proportional to the ratio between ROI and image areas. Experiments show that, for example, 6-8% ratio yields a gain of slightly more than 40% over non-ROI implementation. One of the reasons is that rectification (distortion removal) is still done on the entire image.

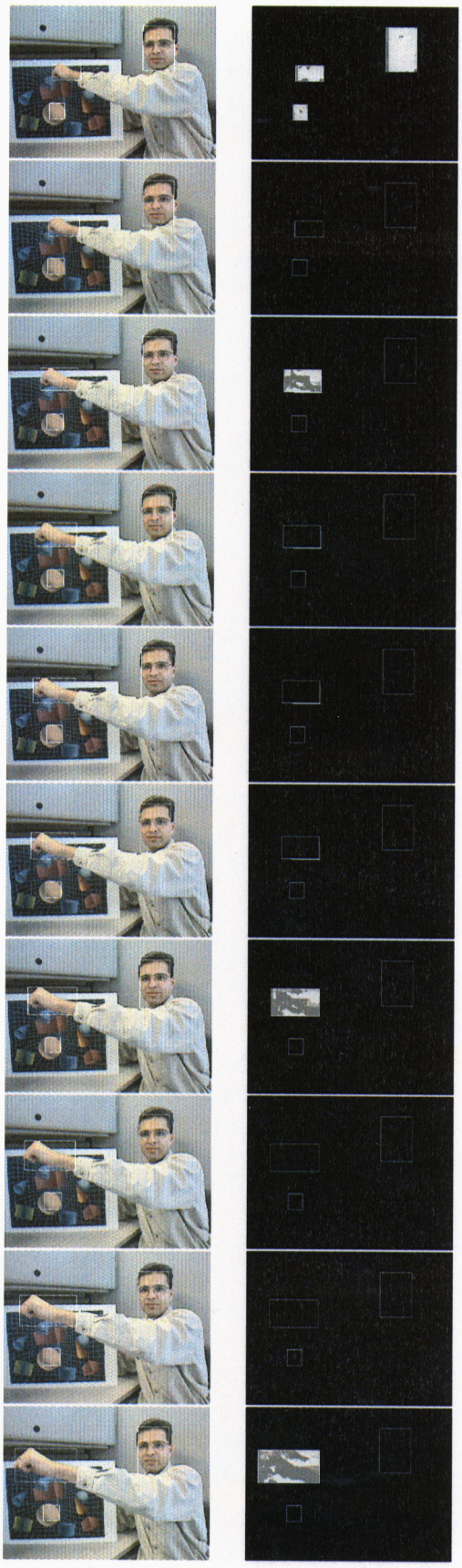


Figure 4: Tracking of skin color regions (left column) and progress in DRRIs (right column) for zoom gesture (hand moving towards the camera).

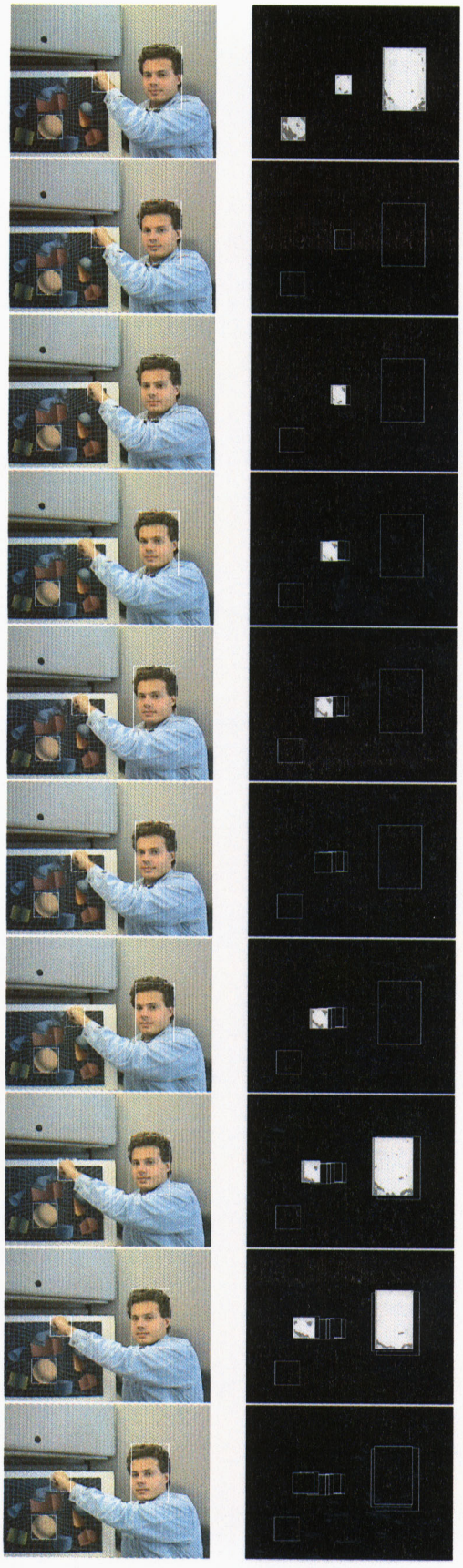
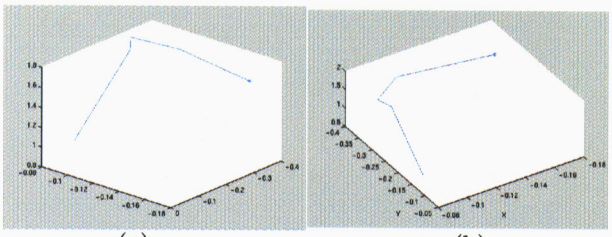


Figure 5: Tracking of skin color regions (left column) and progress in DRRIs (right column) for translation gesture (hand moving side-to-side).

Another reason is that actual region sizes (for the correspondence matching between cameras) also include the number of disparities d searched: for a K by L region, a left-to-right pass is done for a $K(L+d)$ region, and a top-to-bottom pass is performed for a $(K+d)L$ region. Matching on the entire image does not encounter this effect since there is obviously no data outside the image boundaries. Yet, these reasons do not account for the differences between the theoretical and experimental speedup. An implementation maximizing potential advantages of using ROIs can bridge this gap.

Motion trajectories bor both hand movements in a 3-D space are shown in Figures 6(a-b).



(a) (b)

Figure 6: Motion trajectories bor both hand movements in a 3-D space.

4 Additional Aspects of Real-Time Range

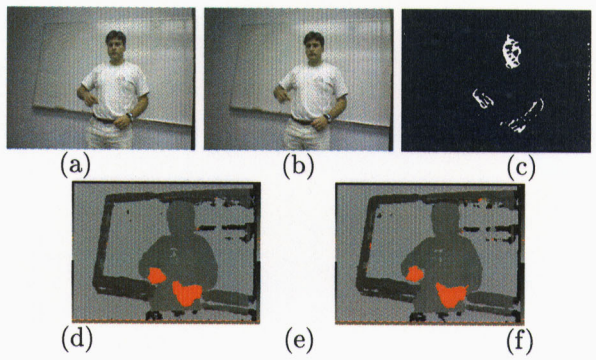
Increased speed of processing can also facilitate a combination of intensity- and range-based input features. Range data enables localized search for specific features, which improves tracking reliability and speed.

Registered range data provides an additional information valuable for segmentation and tracking. Often, an object of interest can be separated from other objects or background by depth alone. In other cases, having fewer artifacts (that could complicate segmentation) in range information compared to intensity data is an important consideration for model matching [16].

Real-time constraints such as temporal correlation produce a possibility of searching within a smaller region, based on the match in the previous frame. For the range image, this involves depth planes immediately surrounding the plane where a hand (or face) was found in the previous frame. Subsequent search in the subset of the intensity data corresponding to these planes produces the position of the body part in the current frame. Therefore, intensity data is thresholded for the certain range and depth. Such combined use of input features produces not only a speedup due to a significant reduction in a search space, but also increased reliability due to a decreased number of false positives that could fall in such space. Rather than processing all pixels, this allows us to select only those pixels with the certain depth, based on the depth of the previously detected region of interest.

Two intensity images from a sequence of the speaking person are shown in Figures 7(a-b). More images are not displayed due to space restrictions. A skin detection algorithm is applied to the intensity data from Figure 7(a). Results of skin thresholding following color segmentation are shown in Figure 7(c). Pixels classified as skin are white. Note that, along with the face and hand information, it picks up up parts of other objects – a curtain on the right and a belt.

Instead of applying the color segmentation again, it is possible to take range data into consideration by selecting one or more depth levels where a region of interest was found (Figure 7(e)). Since the motion between two frames is small, the same level indicates approximate location of the hand in the next frame (red areas in Figure 7(f)). This level, along with the two closest depth levels (before and after), constitute the search space for the current frame (instead of the entire image). Search in the range domain prevented us from considering intensity-based segmentation artifacts (such as a curtain and a belt). Segmentations along intensity and depth channels also can be performed independently and then combined.



(a) (b) (c)

(d) (e) (f)

Figure 7: (a-b) Intensity images of the speaker. (c) Results of skin segmentation and thresholding. (e-f) Range images with selected depth levels.

5 Towards Emulating Redundancies in Biological Vision, and Analogies with Human Attention

One of the aspects of biological vision underutilized in the past for building image understanding systems is data redundancy. Computer vision systems could not afford such extravagant solutions. Combination of real-time intensity and range input data sets proposed by the described method provides possibilities for exploration of other advantages arising from redundancies we can now afford. However, complexity of certain tasks will probably make straight-forward solutions unfeasible in the foreseeable future. Range

processing can be considered one of such tasks. Accuracy demands for many tasks rise faster than hardware improvements. That is why recent precision scanners spend more time on range acquisition and computation than older, less accurate models. For instance, it takes on average more then 30 seconds to acquire and compute range data using a K2T scanner on a SUN SPARC 20, and more than 2 minutes using a Cyberware scanner on Silicon Graphics O2 (considering higher precision achieved with the latter). This makes an interesting case for applying other human perception phenomenons to this problem.

The phenomenon of attention in human vision [4] is a biological solution to the problems of complexity and overabundance of data . It is a means to put the limited resources of the visual system into the right place (and the right orientation) at the right time, and to set the mind in the right context. An important point is the limited amount of available computational power, both in our brain and in the computer (for range processing) that we actually have. On the biological side, this might be the reason for the usefulness of attentional mechanisms. The attentional mechanism: can then be seen as the tool for granting "computational resources" to the tasks, according to their dynamic priorities. Another view of attention is as a mechanism for determining regions of interest in an image (DRRIs in the proposed approach).

Attention includes certain aspects [4] similar to the properties of the described system such as filtering unnecessary data and attending to selected sources only (such as hand motion in the described system), searching for a particular feature (skin color), and expecting something to happen meanwhile attending to empty space (static region outlines in DRRIs).

Another way to schedule the range processing computations is to describe a system of states and operators in such a way that a heuristic evaluation function can guide their application. A sketch of this part of such a system is given in the following:

1) the space of states S includes states that correspond to sets of ROIs;

2) the set of operators includes transformations T between the states in the temporal and spatial senses;

3) there are means of computing the cost of an arc (sl, s2);

4) a current state at any time is represented by a DRRI;

5) a goal-state predicate is defined which returns true when a state represents a completely tracked (and recognized) gesture.

Thus, the process of gesture tracking with real-time range on-demand may be thought of as finding a path through a "regional attention" kind of graph. The path nodes corresponding to states of ROIs. A minimum-cost path from the start state to a goal state is therefore analogous to a maximum likelihood classification of the gesture sequence.

6 Conclusions and Future Work

This paper presented a new approach to a gesture tracking system using real-time range on-demand. The system represents a gesture-controlled interface for interactive visual exploration of large data sets. The paper described a method performing range processing only when necessary and where necessary. This is achieved by a set of filters on the color, motion, and range data. The speedup achieved is between 41% and 54%. The algorithm also includes a robust skin color segmentation insensitive to illumination changes. Selective range processing results in dynamic regional range images (DRRIs). This development is also placed in a broader context of a biological visual system emulation, specifically redundancies and attention mechanisms.

The gesture tracking system described in this paper will be responsible for supplying data manipulation parameters to interactive data exploration and collaborative visualization software. Processing 1 range image for every 4 color images is done at a rate of 10.6 frames per second for a 320x240 image size, and at a rate of 16.5 frames per second for 160x120 images. Therefore, the method is applicable to the real-time processing. More over, since the Triclops library is currently optimized for thread parallel processing, a much greater speedup can be achieved on a dual-processor NT machine (we plan to move the system there in the near future).

Robustness of the approach is achieved with multiple input feature sets. Depth filters are necessary in addition to color, motion and shape filters. Increased speed of processing can also facilitate a combination of intensity- and range-based input features. Range data enables localized search for specific features, which improves tracking reliability and speed.

DRRIs can be included in motion analysis, trajectory computation, gesture recognition, to determine what types of gestures are natural and feasible for robust tracking and interpretation for interactive exploration of large data sets and virtual environments. They can be easily plotted in a 3-D space for movement trajectory parameterization.

Acknowledgments

This was done as a part of SAVAnTS scalable visualization project. I would like to thank Mark Duchaineau, Sam Uselton, Randy Frank, and other members of Visualization Group of the Center for Applied Scientific Computing at the LLNL for their support and suggestions; Erick Cantu-Paz (CASC) for assistance in image acquisition; Regina for the background production; Vladimir Tucakov and Don Murray at the Point Grey Research for technical support;

Dmitry Goldgof at the Computer Science Department of the University of South Florida for invaluable guidance, and Min Shin at the Vision Lab (USF) for the face tracker.

References

[1] J. K. Aggarwal and Q. Cai. Human motion analysis: A review. In *IEEE Nonrigid and Articulated Motion Workshop*, pages 90–102, San Juan, Puerto Rico, June 1997.

[2] A. Azarbayejani and A. Pentland. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. In *Proceedings of International Conference on Pattern Recognition*, Vienna, August 1996.

[3] A. F. Bobick and A. D. Wilson. A state-based approach to the representation and recognition of gesture. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(12):1325–1337, December 1997.

[4] S. Coren, L. M. Ward, and J. T. Enns. *Sensation and Perception.* Harcourt Brace College Publishers, Fort Worth, TX, 1994.

[5] R. Cutler and L. Davis. Real-time periodic motion detection, analysis, and applications. In *Proceedings of IEEE CS Conference on Computer Vision and Pattern Recognition*, volume 2, pages 326–332, Fort Collins, CO, June 1999.

[6] J. W. Davis and A. F. Bobick. The representation and recognition of action using temporal templates. In *Proceedings of IEEE CS Conference on Computer Vision and Pattern Recognition*, pages 928–934, San Juan, PR, June 1997.

[7] D. Gavrila and L. Davis. 3-D model-based tracking of humans in action: a multi-view approach. In *Proceedings of IEEE CS Conference on Computer Vision and Pattern Recognition*, pages 73–80, San Francisco, CA, June 1996.

[8] Point Grey Research Inc. *Triclops Stereo Vision System Version 2.1, User's guide and command reference.* Inc., Point Grey Research, Vancouver, BC, 1996.

[9] Takeo Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka. A stereo machine for video-rate dense depth mapping and its new applications. In *Proceedings of IEEE CS Conference on Computer Vision and Pattern Recognition*, pages 196–202, June 1996.

[10] J. J. Kuch and T.S. Huang. Model-based tracking of self-occluding articulated objects. In *Vision based hand modeling and tracking for virtual teleconferencing and telecollaboration*, pages 666–671, Cambridge, MA, June 1995.

[11] K. Oda, M. Tanaka, A. Yoshida, H. Kano, and Takeo Kanade. A video-rate stereo machine and its application to virtual reality. In *Proceedings of ISPRS '96*, 1999.

[12] V. I. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Trans. on PAMI*, 19(7):677–695, July 1997.

[13] M. W. Powell and R. Murphy. Position estimation of micro-rovers using a spherical coordinate transform color segmenter. In *Proceedings of IEEE Workshop on Photometric Modeling for Computer Vision and Graphics*, Fort Collins, CO, June 1999.

[14] J. M. Rehg and T. Kanade. Visual tracking of high DOF articulated structures: an application to human hand tracking. *Proc. of European Conference on Computer Vision*, 2:35–46, May 1994.

[15] J. Segen and S. Kumar. Shadow gestures: 3D hand pose estimation using a single camera. In *Proceedings of IEEE CS Conference on Computer Vision and Pattern Recognition*, volume 1, pages 479–485, Fort Collins, CO, June 1999.

[16] L. V. Tsap, D. B. Goldgof, and S. Sarkar. Model-based force-driven nonrigid motion recovery from sequences of range images without point correspondences. *Image and Vision Computing Journal*, 17(14):997–1007, November 1999.

[17] L. V. Tsap, D. B. Goldgof, and S. Sarkar. Nonrigid motion analysis based on dynamic refinement of finite element models. *IEEE Trans. on Pattern Analysis and Machine Intelligence.*

[18] S. E. Umbaugh. *Computer Vision and Image Processing.* Prentice-Hall, Englewood Cliffs, NJ, 1998.

[19] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. on PAMI*, 19(7):780–785, July 1997.

[20] Y. Yacoob and M. J. Black. Parameterized modeling and recognition of activities. *Journal of Computer Vision and Image Understanding*, 73(2):232–247, 1999.

[21] M.-H. Yang and N. Ahuja. Recognizing hand gestures using motion trajectories. In *Proceedings of IEEE CS Conference on Computer Vision and Pattern Recognition*, volume 1, pages 466–472, Fort Collins, CO, June 1999.

A Appendix

A.1 Range Computation

The Triclops stereo vision system [8] computes range based on triangulation between cameras. It consists of a three-camera module. Offset in positions of the cameras produces differences in resulting images. These images are compared using square masks to establish correspondences [8]:

$$\min_{d=d_{min}}^{d_{max}} \sum_{i=-\frac{M}{2}}^{\frac{M}{2}} \sum_{j=-\frac{M}{2}}^{\frac{M}{2}} |I_{right}[x+i][y+j] - I_{left}[x+i+d][y+j]| \quad (1)$$

Where d_{min} and d_{max} are the minimum and maximum disparities, m is the mask size, I_{right} and I_{left} are the right and left images, respectively [8]. Since the camera parameters (their relative positions, the focal length and resolution) are fixed, re-calibration is not usually required. According to the multi-baseline stereo theory [9, 11] used in the stereo computation by the system, distance z to the scene point is related to the disparity d, baseline length B and focal length F:

$$z = BF\frac{1}{d} \quad (2)$$

The total amount of computation for stereo processing per frame (required for the Sum of Absolute Differences algorithm) is estimated as [11]:

$$N^2 M^2 d(C-1)P \tag{3}$$

where N^2 is the image size, C is the number of cameras (three for the system used), and P is the number of operations per one square difference calculation.

A.2 SCT Color Space

The main reason SCT became an integral part in numerous applications in the medical imaging [18] and robotics [13] is because it separates the color and brightness information. This allows for a much more reliable segmentation based on the color data which is normally greatly affected by the lighting conditions.

The spherical coordinate transform from the RGB space into a LAB space is defined as [13]:

$$\begin{aligned} L &= \sqrt{R^2+G^2+B^2} \\ \angle A &= cos^{-1}\left[\frac{B}{L}\right] \\ \angle B &= cos^{-1}\left[\frac{A}{L\sin(\angle A)}\right] \end{aligned} \tag{4}$$

where L is the one-dimensional brightness space, and angles A and B determine a two-dimensional color space. Color normalization provides SCTs insensitivity to variations in illumination. L can be viewed as a norm of the vector from the origin to the point in RGB space, A is the angle between the vector and the blue axis, and B is the angle between the red axis and the projection of the vector onto the RG plane (Figure 8(a)). As a result of the transform, a new color space is represented by a color triangle which can be partitioned into the specified number of classes (Figure 8(b)). Greater number of classes improves discrimination. The minimum and maximum A and B values are calculated, defining such areas within the triangle in equal angular increments. The RGB means are defined for each class.

Pixels with certain color properties are found in the image using the minimum distance classifier with Mahalanobis distance. It can be defined as a distance from the feature vector X to the mean vector M_x

$$r = \sqrt{(X-M_x)C_x^{-1}(X-M_x)} \tag{5}$$

where C_x is the covariance matrix for X.

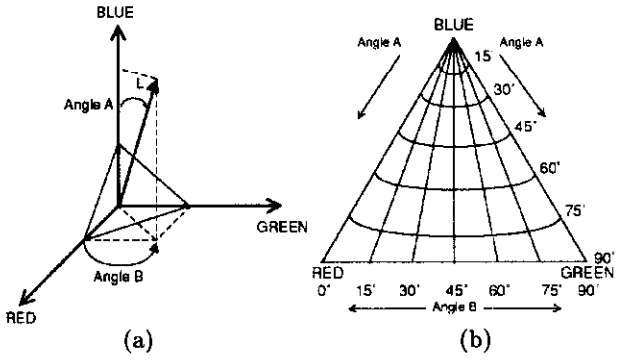


Figure 8: RGB and LAB color spaces: (a) LAB values for a point in RGB space, (b) partitioning of the color triangle into classes.